



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/235,156	01/22/1999	JOSHUA SUSSER	50253-219;P3	5106

7590 01/29/2004

D'Alessandro & Ritchie
P.O. Box 640640
San Jose, CA 95164-0640

EXAMINER

BULLOCK JR, LEWIS ALEXANDER

ART UNIT	PAPER NUMBER
----------	--------------

2126

DATE MAILED: 01/29/2004

31

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary

Application No.

09/235,156

Applicant(s)

SUSSER ET AL.

Examiner

Lewis A. Bullock, Jr.

Art Unit

2126

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133).
- Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) ☒ Responsive to communication(s) filed on 02 January 2004.
- 2a) ☐ This action is **FINAL**. 2b) ☒ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) ☒ Claim(s) 1 and 23-54 is/are pending in the application.
- 4a) Of the above claim(s) _____ is/are withdrawn from consideration.
- 5) ☐ Claim(s) _____ is/are allowed.
- 6) ☒ Claim(s) 1,23-42,44,45,47,48,50,51,53 and 54 is/are rejected.
- 7) ☒ Claim(s) 43,46,49 and 52 is/are objected to.
- 8) ☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☐ The drawing(s) filed on _____ is/are: a) ☐ accepted or b) ☐ objected to by the Examiner.
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. §§ 119 and 120

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
a) ☐ All b) ☐ Some * c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
2. ☐ Certified copies of the priority documents have been received in Application No. _____.
3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).
* See the attached detailed Office action for a list of the certified copies not received.
- 13) ☐ Acknowledgment is made of a claim for domestic priority under 35 U.S.C. § 119(e) (to a provisional application) since a specific reference was included in the first sentence of the specification or in an Application Data Sheet. 37 CFR 1.78.
a) ☐ The translation of the foreign language provisional application has been received.
- 14) ☐ Acknowledgment is made of a claim for domestic priority under 35 U.S.C. §§ 120 and/or 121 since a specific reference was included in the first sentence of the specification or in an Application Data Sheet. 37 CFR 1.78.

Attachment(s)

- 1) ☐ Notice of References Cited (PTO-892) 4) ☐ Interview Summary (PTO-413) Paper No(s). _____
- 2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948) 5) ☐ Notice of Informal Patent Application (PTO-152)
- 3) ☐ Information Disclosure Statement(s) (PTO-1449) Paper No(s) _____ 6) ☐ Other: _____

DETAILED ACTION

Allowable Subject Matter

1. Claims 43, 46, 49, and 52 are objected to as being dependent upon a rejected base claim, but would be allowable if rewritten in independent form including all of the limitations of the base claim and any intervening claims.
2. The following is a statement of reasons for the indication of allowable subject matter: Claims 43, 46, 49, and 52 detail the limitations of their parent claims with the addition that the object instance is associated with a context by recording the name of the context in a header of the object instance, such that information in the header is inaccessible to the one or more program modules. The prior art of record precludes this by having each applet that executes in its respective execution context having complete control over its objects such that all information of the objects is accessible. The cited claims makes Applicant's invention more clearer wherein the objects are owned by the context, and not the applet that instantiated it. Therefore, the cited claims when claimed when rewritten would overcome the prior art of record.

Claim Rejections - 35 USC § 103

1. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

Art Unit: 2126

2. Claims 1 and 23-42, 44, 45, 47, 48, 50, 51, 53, and 54 are rejected under 35 U.S.C. 103(a) as being unpatentable over "Java Card 2.0 Programming Concepts" by SUN.

As to claim 1, SUN teaches a small footprint device (java card / smart card) comprising: at least one processing element (virtual machine / operating system process) (pg. 3, Lifetime of the Virtual Machine) configured to execute groups of program modules (applets) in separate contexts (applet execution context) (pg. 7, "...applets are isolated from each other." Pg. 2, "Each applet is an independent entity with its own state and functionality."; pg. vii, "Applet execution context...), the program modules (applets) comprising zero or more sets of executable instructions (methods) and zero or more sets of data definitions (field contents) grouped as object definitions (classes) (pg. 7, Every object (class instance or array) on the card is owned by the applet which instantiated it...If an applet does not have sharing privileges for an object, any attempt to invoke an instance method or access the object's contents will throw a SecurityException."), each context (applet execution context) comprising a protected object instance space such that at least one of the object definitions (class) is instantiated (class instance) in associated with a particular context (via pg. 7, "...applets are isolated from each other." Pg. 2, "Each applet is an independent entity with its own state and functionality."; pg. vii, "The JCRE keeps track of the currently selected applet as well as the currently active applet and changing contexts accordingly."; pg. 10, "The main task of the install method within the applet is to create and initialize the objects that the applet will need during its lifetime and otherwise prepare itself to be selected

Art Unit: 2126

and accessed by a CAD."); instances of objects (objects instantiated by an applet) (pg. 3, "Every object on the card is owned by the applet which instantiated it. The owning applet always has full privileges to use and modify the object."); and a context barrier (applet firewall) for separating and isolating the contexts (pg. 7, "To create a secure and trusted environment, applets are isolated from each other. An applet firewall prevents one applet from accessing the contents or behavior of objects owned by other applets."), the context barrier (applet firewall) configured for controlling execution of at least one instruction of one of the zero or more sets of instructions (object methods that can be invoked) comprised by a program module (owning applet) based at least in part of whether the at least one instruction is executed for an object instance (object) associated with a first one of the one or more separate contexts (object is part of the owning applet's execution context) and whether the at least one instruction is requesting access to an instance of an object definition (object / class instance) associated with a second one of the said one or more separate contexts (former applet execution context) (via the JCRE, pg. vii, "The JCRE keeps track of the currently selected applet as well as the currently active applet...When a virtual method is invoked on an object, the applet execution context is changed to correspond to the applet that owns that object. When that method returns, the previous context is restored...The applet execution context and sharing status of an object together determine if access to an object is permissible."; pg. 7, "The applet firewall ensures that no other applet may use, access, or modify the contents of an object owned by another applet ..but the applet cannot invoke methods on the object or get or set its contents."), the context barrier further configured to

Art Unit: 2126

prevent the access if the access is unauthorized (pg. 7, "If an applet does not have sharing privileges for an object, any attempt to invoke an instance method or access the object's contents will throw a SecurityException...") and enable the access if the access is authorized (via Unrestricted Sharing or Restricted Sharing) (pg. 8); and a global data structure (JCRE) for permitting one program module (applet) to access information from another program module (applet) by bypassing the context barrier (applet firewall) (pg. 7-8, "However, it is necessary to allow exceptions to this restriction. The JCRE must be able to invoke methods on applets..."; pg. 2, "However, Java Card provides...form of a firewall between applets."). However, SUN does not explicitly mention that the device has memory and that the memory comprises the objects. It is well known to one of ordinary skill in the art that a device has memory and therefore obvious that the device would have memory in order to create the objects and applets.

As to claim 44, SUN teaches the storing object header data (applet identifier), the object header data (applet identifier) comprising information associated with at least one of the instances of objects (via its associated applet); and the controlling execution is based at least in part on the object header data (applet identifier) (pg. 10, "The main task of the install method within the applet is to create and initialize the objects that the applet will need during its lifetime and otherwise prepare itself to be selected and accessed by a CAD....Typically an applet will create various objects, initialize them with predefined values, set some internal state variables, and call the Applet.register method to inform the JCRE that the applet is available for selection. "Selection occurs when the

JCRE receives a SELECT APDU in which the name data matches the AID of the applet. Selection causes an applet to become active, and the applet execution context is adjusted so that only objects belonging to this applet can be accessed.”). It is inherent to the teachings of SUN that the object header data is stored in memory since the JCRE must match a received APDU to every AID corresponding to the registered applets.

As to claim 45, SUN teaches the memory is partitioned into a plurality of memory spaces (execution contexts) with instances of objects (objects) allocated for storage in one of the plurality of storage spaces (execution contexts); and the controlling execution is based at least in part on determining the storage space allocated to an executing object instance (execution context) and an accessed object instance (object) (via JCRE, pg. 10, “Selection occurs when the JCRE receives a SELECT APDU in which the name data matches the AID of the applet. Selection causes an applet to become active, and the applet execution context is adjusted so that only objects belonging to this applet can be accessed.”; pg. vii, “The JCRE keeps track of the currently selected applet as well as the currently active applet...When a virtual method is invoked on an object, the applet execution context is changed to correspond to the applet that owns that object. When that method returns, the previous context is restored.”).

As to claims 32, 47 and 48, reference is made to a method that corresponds to the device of claims 1, 44, and 45 and is therefore met by the rejection of claims 1, 44, and 45 above. However, claim 32 further details the device includes a processing

machine wherein the program modules are executed on. It is obvious that the processing element (virtual machine) of claim 1 is the processing machine of claim 32.

As to claims 34, 50, and 51, refer to claims 1, 44, and 45 for rejection. However, claim 34 further details the creating of the global data structure. It is obvious to one of ordinary skill in the art that since the teachings of SUN have a global data structure that it is created.

As to claims 35, 53, and 54, refer to claims 1, 44, and 45 for rejection. However, claim 35 further details the steps of creating a global data structure; permitting one program module to write information to the global data structure; and having at least one other program module read information from the global data structure thereby bypassing the context barrier. SUN teaches permitting one program module (applet) to write information (method invocation / sending bytes and receiving bytes) to the global data structure (JCRE), and having at least one other program module (applet) read information from the global data structure thereby bypassing the context barrier (applet firewall) (pg. 2, "However, Java Card provides facilities to support more sophisticated scenarios in which multiple applets can discover each other, communicate, and share data in a limited manner, while still maintaining protection from each other in the form of a firewall between applets."; pg. 8, "For method invocation operations, the JCRE remembers the old context, and performs an applet context switch to allow the code in

Art Unit: 2126

the object's applet to function correctly and with expected security restrictions."; pg. 14-16).

As to claims 36 and 37, reference is made to a computer program product that corresponds to the device of claim 1 and is therefore met by the rejection of claim 1 above.

As to claims 38 and 39, refer to claims 36 and 37 for rejection. However, claim 38 further details permitting one program to access information from another program by bypassing a context barrier using a global data structure. SUN teaches permitting one program module (applet) to access information (method invocation / sending bytes and receiving bytes) from another program module (applet) using the global data structure (JCRE) (pg. 2, "However, Java Card provides facilities to support more sophisticated scenarios in which multiple applets can discover each other, communicate, and share data in a limited manner, while still maintaining protection from each other in the form of a firewall between applets."; pg. 8, "For method invocation operations, the JCRE remembers the old context, and performs an applet context switch to allow the code in the object's applet to function correctly and with expected security restrictions."; pg. 14-16).

As to claim 40, reference is made to a computer wave that corresponds to the device of claim 1 and is therefore met by the rejection of claim 1 above.

As to claim 41, refer to claim 40 for rejection. However, claim 41 further details permitting one program to access information from another program using the one global data structure. SUN teaches permitting one program module (applet) to access information (method invocation / sending bytes and receiving bytes) from another program module (applet) using the global data structure (JCRE) (pg. 2, "However, Java Card provides facilities to support more sophisticated scenarios in which multiple applets can discover each other, communicate, and share data in a limited manner, while still maintaining protection from each other in the form of a firewall between applets."; pg. 8, "For method invocation operations, the JCRE remembers the old context, and performs an applet context switch to allow the code in the object's applet to function correctly and with expected security restrictions."; pg. 14-16).

As to claim 42, refer to claim 1 for rejection. However, claim 42 further details the transmitting of a code over a network wherein the code is instructions for implementing a global data structure for bypassing a context barrier. It is obvious to one of ordinary skill in the art that the firewall and the JCRE has program code in order to function on the java card system. However, SUN does not teach that the code is sent over a communications link. It is well known to one of ordinary skill in the art that computer code is downloaded from a developer system or server system to an implementation system or client system. Therefore, it is obvious to one skilled in the art

at the time of the invention that the carrier wave code of the firewall and JCRE is shipped or downloaded from a server system to a client system to be implemented.

As to claims 23-26, SUN teaches that each applet has its own context (Applet execution context) (pg. vii, Terminology) and that the applets are separated by an applet firewall (pg. 7) and an applet can access another applet and its object by the JCRE (pg. 7-8). It is well known to one of ordinary skill in the art that an execution context has a memory space or name space. Therefore, it is obvious that the applets have their separate memory spaces or name spaces for each applets execution. It is also obvious that since an applet can access another applet via the JCRE that the multiple applets can access one another through the JCRE when allowed.

As to claims 27-31, SUN teaches the context barrier (applet firewall) prevents access from a principle (applet) in one context to an object in a different context (applet) (pg. 7, Applet Isolation and Object Sharing, "An applet firewall prevent one applet from accessing the contents or behavior of objects owned by other applets."; pg. 2, Multiple Applets, "However, Java Card provides...in which multiple applets can discover each other, communicate, and share data in a limited manner, while still maintaining protection from each other in the form of a firewall between applets."). It is obvious that since the context barrier prevents object access to an applet not owning the objects (pg. 7) that the context barrier enforces a security check on the applet accessing of the object. It is also obvious that the security check involves name / memory space

agreement since the applet can only access objects within its execution context and it is well known to one of ordinary skill in the art that an execution context has a memory space or name space.

As to claims 33, SUN teaches the applet firewall prevents one applet from accessing the contents or behavior of objects owned by other applets (pg. 7, Applet Isolation and Object Sharing) and that when one applet invokes another applet's objects, the JCRE performs applet context switch to allow the code in the objects applet to perform the method invocation operation (pg. 8, Applet Isolation and Object Sharing). Therefore, it would be obvious that the firewall prevents access from a principal to an object unless they are on the same context and unless they access the JCRE for allow the access.

Response to Arguments

3. Applicant's arguments filed 1/2/04 have been fully considered but they are not persuasive. Applicant argues that in Sun reference does not teach the cited amendments to the claims. The examiner disagrees and refers to the rejection in illustrating his point. Applicant then argues that the invention describes objects as being owned by a context. Specifically as defined in claim 1, "each context comprising a protected object instance space such that at least one of the object definitions is instantiated in associated with a particular context". However, the passage only details that each context comprises a protected object instance space and that the object

definitions are instantiated in an associated particular context. Sun teaches that each applet executes in an applet execution context, the applet creates its own objects, and the applet firewall protects one applet from another. Therefore, the applet execution context is a protected object space for an applet to create objects. Only dependent claims 43, 46, 49, and 52 detail that a context is associated with an object such that the program module has no control over which object it controls access to. These claims seem to depict what Applicant is centrally arguing to overcome the reference of Sun and as detailed by the Examiner above would make the invention allowable.

Conclusion

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Lewis A. Bullock, Jr. whose telephone number is (703) 305-0439. The examiner can normally be reached on Monday-Friday, 8:30 am - 5:00 pm.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Meng An can be reached on (703) 305-9678. The fax phone number for the organization where this application or proceeding is assigned is (703) 872-9306.

Any inquiry of a general nature or relating to the status of this application or proceeding should be directed to the receptionist whose telephone number is (703) 305-0286.

Lewis A. Bullock Jr.